

RECEIVED  
CENTRAL FAX CENTER

MAY 27 2008

Appl. No. 10/712,463

Amdt. dated May 27, 2008

Reply to Final Office Action of March 27, 2008

## AFTER FINAL - EXPEDITED PROCEDURE

REMARKS

Claims 1 to 78 were pending in the application at the time of the advisory action. Claims 16, 18, 35, 37, 54, 56, 73, and 75 remain rejected for obviousness type double-patenting. Claims 1 to 78 remain rejected as obvious.

Claims 16, 18, 35, 37, 54, 56, 73 and 75 stand rejected for obviousness-type double-patenting in view of U.S. Patent No. 7,107,581, hereinafter referred to as the '581 patent.

Enclosed herein is a Terminal Disclaimer to Obviate a Double Patenting Rejection Over a Prior Patent Including a Statement Under 37 C.F.R. § 3.73(b) with respect to the '581 Patent. Authorization to charge the appropriate fee for the Terminal Disclaimer is also included. Therefore, Applicants respectfully submit that the obviousness-type double-patenting rejection of Claims 16, 18, 35, 37, 54, 56, 73 and 75 in view of the '581 patent is rendered moot. Applicants respectfully request reconsideration and withdrawal of the obviousness-type double-patenting rejection of each of Claims 16, 18, 35, 37, 54, 56, 73 and 75 in view of the '581 patent.

Claims 16, 18, 35, 37, 54, 56, 73 and 75 stand rejected for obviousness-type double-patenting in view of U.S. Patent No. 7,207,037, hereinafter referred to as the '037 patent.

Enclosed herein is a Terminal Disclaimer to Obviate a Double Patenting Rejection Over a Prior Patent Including a Statement Under 37 C.F.R. § 3.73(b) with respect to the '037 Patent. Authorization to charge the appropriate fee for the Terminal Disclaimer is also included. Therefore, Applicants respectfully submit that the obviousness-type double-patenting rejection of Claims 16, 18, 35, 37, 54, 56, 73 and 75 in view of the '037 patent is rendered moot. Applicants respectfully request reconsideration and withdrawal of the obviousness-type double-patenting rejection of each of Claims 16, 18, 35, 37, 54, 56, 73 and 75 in view of the '037 patent.

Page 2 of 9

Appl. No. 10/712,463  
Amdt. dated May 27, 2008  
Reply to Final Office Action of March 27, 2008

### AFTER FINAL - EXPEDITED PROCEDURE

Claims 1 to 78 stand rejected under 35 U.S.C. 103(a) as being unpatentable over U.S. Patent No. 5,740,441, hereinafter referred to as Yellin, in view of U.S. Patent No. 6,308,317, hereinafter referred to as Wilkinson.

Applicants respectfully traverse the obviousness rejection of Claim 1. Applicants respectfully note that the MPEP requires that both the claims and the references be considered as a whole. "Ascertaining the differences between the prior art and the claims at issue requires interpreting the claim language, and considering both the invention and the prior art references as a whole." MPEP § 2143.02, 8<sup>th</sup> Ed., Rev. 6, pg. 2100-123 (September 2007).

Claim 1 is directed at "A method for arithmetic expression optimization" and recites specific elements performed in the optimization and in particular, validating, converting, and matching elements. The rejection, as previously pointed out, cited portions of Yellin against each of these elements based upon misinterpretations and mischaracterizations of Yellin. Such misinterpretations and mischaracterizations do not comply with the "as a whole" requirements of the MPEP and demonstrate that a prima facie obviousness rejection that complies with the requirements of the MPEP has not been made.

First, Yellin fails to teach or suggest "reconfigure said first instruction to operate as one operand of a second type, said second type smaller than said first type," as asserted in the final rejection. As explained by Yellin,

The present invention verifies the integrity of computer programs written in a bytecode language, commercialized as the JAVA bytecode language, which uses a restricted set of data type specific bytecodes. All the available source code bytecodes in the language either (A) are stack data consuming bytecodes that have associated data type restrictions as to the types of data that can be processed by each such bytecode, (B) do not utilize stack data but affect the stack by either adding data of known data type to the stack or by removing data from the stack

GUNNISON MCKAY &  
HODGSON, L.L.P.  
Garden West Office Plaza  
1900 Garden Road, Suite 270  
Menlo Park, CA 94025  
(415) 655-0880  
Fax (415) 655-0188

Appl. No. 10/712,463

Amdt. dated May 27, 2008

Reply to Final Office Action of March 27, 2008

MAY 27 2008

## AFTER FINAL - EXPEDITED PROCEDURE

without regard to data type, or (C) neither use stack data nor add data to the stack.

The present invention provides a verifier tool and method for identifying, prior to execution of a bytecode program, any instruction sequence that attempts to process data of the wrong type for such a bytecode or if the execution of any bytecode instructions in the specified program would cause underflow or overflow of the operand stack, and to prevent the use of such a program.  
(Emphasis Added)

Yellin, Col. 1, line 57 to Col. 2, line 7. Thus, Yellin taught that if an instruction sequence attempts to process data of the wrong type or if execution of any bytecode instructions cause overflow or underflow, use of the program is prevented. Yellin does not teach any action toward correcting the problem, but instead specifically stated that the use of the program is prevented. Accordingly, taken as a whole, Yellin teaches away from any correction of the problems detected in the verification process by teaching that when a problem is detected, use of the program is prevented.

Applicants have pointed out multiple instances in Yellin of similar teachings. Contrary to the statements in the final rejection, considering Yellin as a whole is not attacking the reference individually, but rather as directed by the MPEP, "A prior art reference must be considered in its entirety, i.e., as a whole, including portions that would lead away from the claimed invention." MPEP § 2141.02, 8<sup>th</sup> Ed., Rev. 6, p. 2100-126, (Sept. 2007).

Claim 1 recites in part:

converting said first instruction to a second instruction configured to operate on at least one operand of a second type, said second type smaller than said first type, said converting based at least in part on the relative size of said first type and said second type, wherein said second instruction is different from said first instruction

GUNNISON, McKay &  
HODGSON, LLP  
Garden West Office Plaza  
1900 Garden Road, Suite 220  
Menlo Park, CA 94025  
(415) 655-0888  
Fax (415) 655-0888

Appl. No. 10/712,463

Amdt. dated May 27, 2008

Reply to Final Office Action of March 27, 2008

### AFTER FINAL - EXPEDITED PROCEDURE

The rejection of this element stated in part:

reconfigure said first instruction to operate as one operand of a second type, said second type smaller than said first type (e.g. overflow - Fig. 4C, 4D; updated ... modified ... by the current instructions - col. 10, line 58 to col. 11, line 6; and improving execution time efficiency - col. 5, line 65 to col. 6, line 4), said reconfiguring based at least in part on the relative size of said first type and said second type (Note: overflow analysis for each successor instruction reads on size of destination place in stack for a successor operand being smaller to take required size of upper instruction in the data flow - see Fig. 4A-B; different data types -- col. 21, lines 16-38);

This is a mischaracterization of the teaching of Yellin. Yellin does not reconfigure any instruction in the bytecode program, as asserted in the rejection, and instead explicitly teaches that when a problem is encountered, the problem is noted and the verification processing aborted.

Specifically, with respect to the cited Fig. 4C,

If a mismatch is detected (458) between the stored operand information in the popped entry of the virtual stack 302 and the data type requirements of the currently selected instruction, then a message is generated (462) identifying the place in the bytecode program where the mismatch occurred. The verifier will then set the VerificationSuccess flag to False and abort (456) the verification process. This completes the stack pop verification process. (Emphasis Added.)

Yellin, Col. 9, lines 53 to 60.

With respect to the cited Fig. 4D,

Referring to FIG. 4D, if the currently selected instruction pushes data onto the stack (470), the stack counter is inspected (472) to determine whether there is sufficient room in the stack to store the data the selected instruction will push onto the stack. If the operand stack has insufficient room to store the data to

GUNNISON, MCKAY &  
HODGSON, LLP  
Carden West Office Plaza  
1500 Garden Road, Suite 770  
Menlo Park, CA 94025  
(831) 455-0888  
Fax (831) 455-0888

Appl. No. 10/712,463  
Amdt. dated May 27, 2008  
Reply to Final Office Action of March 27, 2008

#### AFTER FINAL - EXPEDITED PROCEDURE

be pushed onto the stack by the current instruction (472), that is called a stack overflow, in which case an error signal or message is generated (474) identifying the place in the program that the stack underflow was detected. In addition, the verifier will then set the VerificationSuccess flag to False and abort (476) the verification process. (Emphasis Added.)

Yellin, Col. 9, line 61 to Col. 10, line 5.

Thus, contrary to the interpretation in the rejection, Yellin teaches away from reconfiguring any instruction when an instruction is improper as determined in the verification process and simply aborts the verification process. As noted above, when the verification process is aborted, Yellin taught that execution of the bytecode program is prevented.

Thus, according to Yellin, the bytecode program can be executed by the interpreter only when no problems are detected by the verifier. Thus, the statement in Yellin at Col. 5, line 65 to Col. 6, line 4 cited in the rejection

Use of the bytecode verifier 120 in accordance with the present invention enables verification of a bytecode program's integrity and allows the use of an interpreter 122 which does not execute the usual stack monitoring instructions during program execution, thereby greatly accelerating the program interpretation process.

when taken in context does not support the interpretation in the rejection.

The portions of Yellin relied upon in the rejection are associated with the bytecode verifier, while the efficiency is associated with the execution process of the interpreter. The advantage in efficiency applies only if the verification process is successful. Specifically, as taught by Yellin,

If all the methods are successfully verified (564) an object instance of the object class is generated, and the bytecode interpreter 122 is invoked (570) to execute the user requested procedure, which is typically called a

GUNNISON, MCKAY &  
HODGSON, L.L.P.  
Garden West Office Plaza  
1900 Garden Road, Suite 220  
Monterey, CA 93940  
(415) 655-0880  
Fax (415) 655-0888

Appl. No. 10/712,463  
Amdt. dated May 27, 2008  
Reply to Final Office Action of March 27, 2008

### AFTER FINAL - EXPEDITED PROCEDURE

method. The bytecode interpreter of the present invention does not perform (and does not need to perform) any operand stack overflow and underflow checking during program execution and also does not perform any data type checking for data stored in the operand stack during program execution. These conventional stack overflow, underflow and data type checking operations can be skipped by the present invention because the verifier has already verified that errors of these types will not be encountered during program execution.

Yellin, Col. 14, line 66 to Col. 15, line 11.

Thus, not only does Yellin fail to teach any reconfiguration of an instruction in the bytecode program, but also the efficiency is obtained only when the bytecode program is correct and so would not require any reconfiguration. Accordingly, the rejection extracts teachings from Yellin and applies the teachings contrary to Yellin taken as a whole.

As noted above, Yellin taught two outcomes. If a problem was detected, verification was aborted and execution of the program was prevented. If no problems were detected and verification completed successfully, i.e., the bytecode program was executed by an interpreter.

The rejection proposes to modify Yellin based upon Wilkinson. However, to modify Yellin to change instructions in the bytecode program changes the principles of operation of Yellin. If the program was verified successfully, no changes would be necessary. If the program could not be verified, execution was prevented and verification aborted. Thus, if Yellin is modified to change an instruction instead of aborting and preventing execution, the modification changes the principles of operation of Yellin. The MPEP directs:

If the proposed modification or combination of the prior art would change the principle of operation of the prior art invention being modified, then the teachings of the references are not sufficient to render the claims *prima facie* obvious.

GUNNISON, MCKAY &  
HODGSON, LLP.  
Garden Wood Office Plaza  
1900 Garden Road, Suite 210  
Menlo Park, CA 94025  
(415) 655-0880  
Fax (415) 655-0888

Appl. No. 10/712,463  
Amdt. dated May 27, 2008  
Reply to Final Office Action of March 27, 2008

**AFTER FINAL - EXPEDITED PROCEDURE**

MPEP § 2143.01, VI., 8<sup>th</sup> Ed., Rev. 6, p. 2100-141 (Sept. 2007).

Thus, the teaching of Yellin and Wilkinson are not sufficient to render the claims prima facie obvious. Applicants respectfully request reconsideration and withdrawal of the obviousness rejection of Claim 1.

Claims 2 to 19 depend from Claim 1 and so distinguish over the combination of references for at least the same reasons as Claim 1. Applicants request reconsideration and withdrawal of the obviousness rejection of each of Claims 2 to 19.

Claims 20, 39, 58, 77 and 78 each include limitations similar to those of Claim 1. Accordingly, the above remarks with respect to Claim 1 are applicable to each of these claims and are incorporated herein by reference. Applicants request reconsideration and withdrawal of the obviousness rejection of each of Claims 20, 39, 58, 77 and 78.

Claims 21 to 38 depend from Claim 20 and so distinguish over the combination of references for at least the same reasons as Claim 20. Applicants request reconsideration and withdrawal of the obviousness rejection of each of Claims 21 to 38.

Claims 40 to 57 depend from Claim 39 and so distinguish over the combination of references for at least the same reasons as Claim 39. Applicants request reconsideration and withdrawal of the obviousness rejection of each of Claims 40 to 57.

Claims 59 to 76 depend from Claim 58 and so distinguish over the combination of references for at least the same reasons as Claim 58. Applicants request reconsideration and withdrawal of the obviousness rejection of each of Claims 59 to 76.

GUNNISON, MCKAY &  
HODGSON, L.L.P.  
Cordis West Office Plaza  
1000 Cordis Road, Suite 220  
Menlo Park, CA 94025  
(831) 655-0881  
Fax (831) 655-0888

RECEIVED

MAY 27 2008

Appl. No. 10/712,463

Amdt. dated May 27, 2008

Reply to Final Office Action of March 27, 2008

## AFTER FINAL EXPEDITED PROCEDURE

Claims 1 to 78 remain in the application. For the foregoing reasons, Applicant(s) respectfully request allowance of all pending claims. If the Examiner has any questions relating to the above, the Examiner is respectfully requested to telephone the undersigned Attorney for Applicant(s).

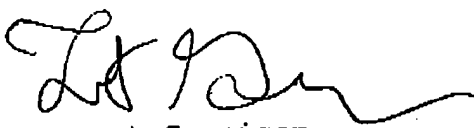
Respectfully submitted,

## CERTIFICATE OF TRANSMISSION

I hereby certify that this correspondence is being facsimile transmitted to the U.S. Patent and Trademark Office, Fax No. (571) 273-8300, on May 27, 2008.

  
Mona Marshall

May 27, 2008  
Date of Signature

  
Forrest Gunnison  
Attorney for Applicant(s)  
Reg. No. 32,899